

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-244957

(43)Date of publication of application : 07.09.2001

(51)Int.Cl.  
H04L 12/46  
H04L 12/28  
G06F 13/00  
H04L 12/66  
H04L 12/56  
H04M 3/00

(21)Application number : 2000-051334

(71)Applicant : FUJITSU LTD  
NTT DOCOMO INC

(22)Date of filing : 28.02.2000

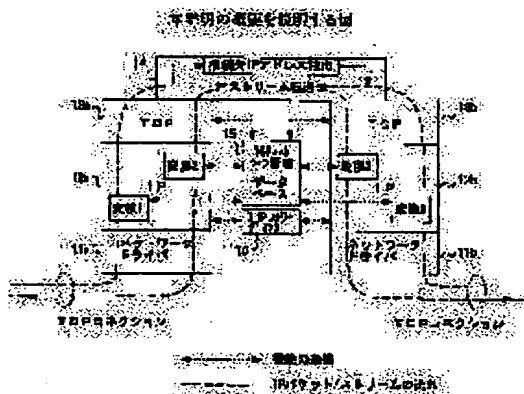
(72)Inventor : TOKUYO MASANAGA  
NAKAGAWA ITARU  
CHIKUMA SATOSHI  
FUJINO SHINJI  
TANIGUCHI TETSUYA  
HISANAGA TAKANORI  
CHIKADA MICHIMASU  
KUWATA DAISUKE

## (54) IP ROUTER DEVICE WITH TCP TERMINATING FUNCTION AND MEDIUM

## (57)Abstract:

PROBLEM TO BE SOLVED: To provide an IP router device equipped with a function for terminating a TCP connection, with which global connectivity is guaranteed.

SOLUTION: When plural IP packets forming a TCP connection to be repeated pass through a router device 1, converting means 1 and 4 cooperate with a TCP connection managing data base 15 and rewrite one part of IP address information and port number information in the packets according to prescribed rules. A stream linking means 14 extracts information showing the original connection destination of the TCP connection from the packets and links two TCP connections through a stream. Converting means 2 and 3 cooperate with the managing data base 15 and rewrite one part of IP address information and port number information in the plural IP packets according to prescribed rules so as to form the TCP connection to the original connection destination.



## LEGAL STATUS

[Date of request for examination] 05.06.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration].

[Date of final disposal for application]

[Patent number] 3494610

[Date of registration] 21.11.2003

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号  
特開2001-244957  
(P2001-244957A)

(43)公開日 平成13年9月7日(2001.9.7)

(51)Int.Cl. <sup>7</sup>	識別記号	F I	テーマコード*(参考)
H 0 4 L 12/46		G 0 6 F 13/00	3 5 3 C 5 B 0 8 9
12/28		H 0 4 M 3/00	B 5 K 0 3 0
G 0 6 F 13/00	3 5 3	H 0 4 L 11/00	3 1 0 C 5 K 0 3 3
H 0 4 L 12/66		11/20	B 5 K 0 5 1
12/56			1 0 2 A 9 A 0 0 1

審査請求 未請求 請求項の数 6 O L (全 16 頁) 最終頁に続く

(21)出願番号 特願2000-51334(P2000-51334)

(22)出願日 平成12年2月28日(2000.2.28)

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番  
1号

(71)出願人 392026693

株式会社エヌ・ティ・ティ・ドコモ

東京都千代田区永田町二丁目11番1号

(72)発明者 徳世 雅永

神奈川県川崎市中原区上小田中4丁目1番  
1号 富士通株式会社内

(74)代理人 100100930

弁理士 長澤 俊一郎 (外1名)

最終頁に続く

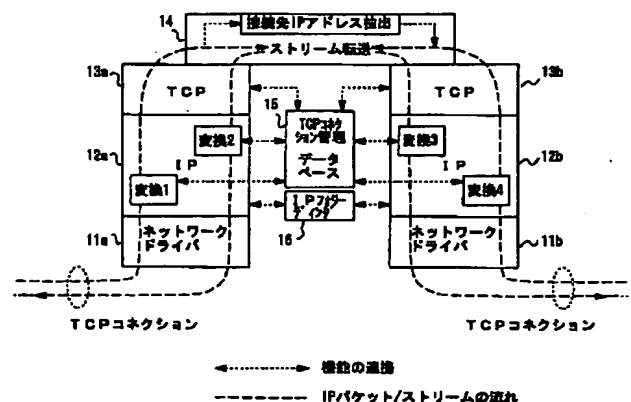
(54)【発明の名称】 TCP終端機能付きIPルータ装置および媒体

(57)【要約】

【課題】 グローバルコネクティビティを保証したTCPコネクションを終端させる機能を備えたIPルータ装置を実現すること。

【解決手段】 中継しようとするTCPコネクションを形成する複数のIPパケットがルータ装置1を通過しようとする際に、変換手段1、4は、TCPコネクション管理データベース15と連携し、パケット内のIPアドレス情報およびポート番号情報の一部を所定の規則に従い書き換える。ストリーム連結手段14は、パケットからTCPコネクションの本来の接続先を示す情報を抽出し、2つのTCPコネクションをストリームで連結する。変換手段2、3は、上記管理データベース15と連携して本来の接続先へのTCPコネクションを形成するように複数のIPパケット内のIPアドレス情報およびポート番号情報の一部を所定の規則に従い書き換える。

本発明の概要を説明する図



## 【特許請求の範囲】

【請求項 1】 TCPコネクションを終端させる機能を備えた複数の異なる IP ネットワークを接続するルータ装置であって、

上記ルータ装置は、中継しようとする TCPコネクションを形成する複数の IP パケットが前記ルータ装置を通過しようとする際に、パケット内の IP アドレス情報およびポート番号情報の一部を所定の規則に従い書き換える第 1 の変換手段と、

前記 TCPコネクションの本来の接続先を示す情報を抽出し、前記ルータからその接続先へ TCPコネクションを生成し、前記 2 つの TCPコネクションをストリームで連結する手段と、

上記本来の接続先への TCPコネクションについて、それを形成する複数の IP パケット内の IP アドレス情報およびポート番号情報の一部を所定の規則に従い書き換える第 2 の変換手段とを備え、

上記第 1、第 2 の変換手段は、上記 2 つの TCPコネクションを 1 組として取り扱い、上記 1 組の TCPコネクションに対して一意の識別番号を付与してデータベースに格納し、該データベースに格納された一意の識別番号により、2 つの TCPコネクションを管理することを特徴とする IP ルータ装置。

【請求項 2】 上記第 1、第 2 の変換手段は、TCP スタックからの TCPコネクションの終了の通知を受けて、上記データベースに格納している識別番号の情報を作成、消去もしくは、更新することを特徴とする請求項 1 の IP ルータ装置。

【請求項 3】 上記第 1、第 2 の変換手段は、TCP 以外の IP パケットについては、IP パケット内の情報の書き換えを行わず、所定のネットワークにルーティングすることを特徴とする請求項 1 または請求項 2 のルータ装置。

【請求項 4】 上記第 1、第 2 の変換手段は、所定のコネクション数を超えた新たな TCPコネクションの IP パケットについて、IP パケット内の情報の書き換えを行わず、所定のネットワークにルーティングすることを特徴とする請求項 1、2 または請求項 3 の IP ルータ装置。

【請求項 5】 上記第 1、第 2 の変換手段は、所定の型をパケット内のヘッダ情報として持つ ICMP パケットについて、その IP アドレス情報およびポート番号情報、および、パケットデータ内に含まれる TCP / IP のヘッダ情報の一部を書き換えることを特徴とする請求項 1、2、3 または請求項 4 のルータ装置。

【請求項 6】 TCPコネクションを終端させる機能を備えた複数の異なる IP ネットワークを接続するプログラムを記録した記録媒体であって、

上記プログラムは、中継しようとする TCPコネクションを形成する複数の IP パケットが前記ルータ装置を通

過しようとする際に、パケット内の IP アドレス情報およびポート番号情報の一部を所定の規則に従い書き換え、

前記 TCPコネクションの本来の接続先を示す情報を抽出し、前記ルータからその接続先へ TCPコネクションを生成し、前記 2 つの TCPコネクションをストリームで連結し、

上記本来の接続先への TCPコネクションについて、それを形成する複数の IP パケット内の IP アドレス情報およびポート番号情報の一部を所定の規則に従い書き換え、

上記 2 つの TCPコネクションを形成する複数の IP パケット内の IP アドレス情報およびポート番号情報の一部を書き換えるに際し、上記 2 つの TCPコネクションを 1 組として取り扱い、上記 1 組の TCPコネクションに対して一意の識別番号を付与してデータベースに格納し、該データベースの格納された一意の識別番号により、2 つの TCPコネクションを管理することを特徴とする複数の異なる IP ネットワークを接続するプログラムを記録した記録媒体。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 従来、インターネットはイーサネット（登録商標）を中心とした有線で構成されていたが、近年、インターネットを構成するネットワークが多様化し、携帯電話、PHS（登録商標）、さらには無線パケット装置などを利用した IP ネットワークが広く使われるようになってきている。無線ネットワークを含むインターネットにおいて、与えられた帯域を有効に利用するためには、有線と無線のように異なる性質を持つネットワークを結合しても、それらを跨るホスト間のコネクションにおいて効率的な伝送レートが得られる IP ルータが必要とされている。本発明は、上記インターネットプロトコル（IP）を基盤としたコンピュータネットワークにおいて、異なるネットワークを結合し IP パケットをネットワーク間で橋渡しする TCPコネクションを終端させる機能を備えた IP ルータ装置に関する。

## 【0002】

【従来の技術】 有線のネットワークでは、IP パケットの紛失や遅延はネットワークを結ぶ IP ルータ内での輻輳によるものがほとんどであった。一方、無線のネットワークでは、物理層すなわち電波として変調されたデータが、電波の減衰、干渉や物体による遮蔽などによる状況変化の影響を受けてデータが正しく伝送されないことによって、IP パケットの損失や遅延が発生する。現在インターネットでは、信頼性のあるデータ伝送を行なうために、TCP（Transmission Control Protocol）というプロトコルが標準として使われている。

【0003】 しかしながら、TCP は有線のネットワー

## 3

クで起こる輻輳を主たるパケット損失／遅延の原因としており、無線ネットワークではTCPを制御するパラメータおよびアルゴリズムを無線用に最適化しないと、効率的な伝送レートが得られないことが知られている。これに対して、TCP層でTCPコネクションを一度終端し、性質の異なるネットワークに対して別々のTCPコネクションを持つことによって、それぞれに適した制御パラメータやアルゴリズムを使用し、効率的な伝送レートを得るアプリケーションゲートウェイが提案されている。

## 【0004】

【発明が解決しようとする課題】従来型IPルータ装置の構成を図16に、また従来型TCP終端装置の構成を図17に示す。IPルータ装置は、図16に示すようにネットワークドライバ11a、11b、IPスタック12a、12b、IPフォワーディング16から構成され、図16に示すようにIPパケットをネットワーク間で受け渡す。図18にIPパケットのヘッダ構成を、図19にTCPパケットのヘッダ構成を、図20にICMPパケットのヘッダ構成を示す。IPパケットのヘッダ部には図18に示すように送信元のIPアドレスを示すsrc-IP (SOURCE IP ADDRESS)、接続先のアドレスを示すdst-IPアドレス (DESTINATION IP ADDRESS) が記述され、TCPパケットのヘッダ部には図19に示すように送信元のポート番号であるsrc-ポート番号 (SOURCE PORT)、接続先のポート番号であるdst-ポート番号 (DESTINATION PORT) が記述される。また、ICMPパケットのヘッダ部にはTYPE (型) と、IPヘッダと最初の64ビットのデータグラムが記述される。

【0005】図21に上記従来型のIPルータ装置のIPスタック12a、12bにおけるIPパケットの入力処理のフローを示し、図22にIPルータ装置のIPスタック12a、12bにおけるIPパケットの出力処理のフローを示す。入力処理において、IPスタックは図21に示すようにIPパケットのdst-IPアドレスが自ホスト宛か判断し、自ホスト宛の場合には自ホストのTCP/UDPスタックにパケットを引き渡す。また、自ホスト宛でない場合には、IPフォワーディング経由でIPパケットを出力処理に引き渡す。また、出力処理において図22に示すようにdst-IPアドレスもしくはそのサブネットワークでルーティングテーブルを検索し、該ルーティングテーブルに従い、ネットワークドライバにIPパケットを引き渡す。

【0006】IPルータ装置は、上記のようにIPパケットをネットワーク間で受け渡しているだけなので、エンドホストでは基本的にルータの存在を関知しない。このルータ装置を介した任意のホスト間においては、一意にIPパケットを到達させることが可能であり、IPパケットヘッダ内のsrc-IPアドレスおよびdst-

## 4

IPアドレスの情報はなんら変化しない。また、エンド-エンドの経路上に位置するルータの構成や数が変化しても、影響を受けない。すなわち、インターネットは、その性質の一つとして、各ホストがグローバルに一意なIPアドレスを持ち、そのホストから任意のホストへ通信ができ、かつ、任意のホストからそのホストへ通信ができる。これを「エンド-エンドのグローバルコネクティビティの保証」という。

【0007】一方、従来型TCP終端装置 (アプリケーションゲートウェイ) は、図17に示すように、ネットワークドライバ11a、11bと、IPスタック12a、12bとTCPスタック13a、13bとストリーム転送手段14、および接続情報を保持するデータベース20から構成される。図23に従来型のTCP終端装置 (アプリケーションゲートウェイ) におけるストリーム転送機能のフローチャートを示す。同図に示すようにストリーム転送機能は、サーバソケットとしてTCPコネクションの接続を待ち (ステップS1)、クライアントとTCPコネクションを確立する (ステップS2)。ついで、ストリーム内のデータから接続先のIPアドレスとポート番号を取得し (ステップS3)、クライアントソケットとして、上記接続先にTCPコネクションを確立する (ステップS4)。そして、ストリームが継続しているかを判断し (ステップS5)、継続していない場合には処理を終了する。処理が継続している場合には、クライアントとのストリームから一定量以下のデータを読み出し (ステップS6)、サーバとのストリームへ前記データを書き出す (ステップS7)。

【0008】上記したTCP終端装置 (アプリケーションゲートウェイ) においては、以下に示す理由から前記した「グローバルコネクティビティの保証」ができないという問題点を持つ。

(a) プロトコル依存アプリケーションゲートウェイの多くは、特定のプロトコルのみにに対してTCP終端を行なう。そのためサポート外のアプリケーションはIPパケットもしくはTCPストリームが流れない。

(b) エンド-エンド情報の非保存ゲートウェイ自身がTCPを終端するために、ゲートウェイのホストが、それぞれのエンドホストに対して対向しているエンドホストとして見える。つまり、本来のエンドホスト同士がお互いを自分のエンドホストとして認識できなくなる。

【0009】TCP終端装置 (アプリケーションゲートウェイ) を実現する上で重要なことの一つに接続先の情報をクライアントホストから「何らかの手段」で通知してもらう必要がある。具体的な例として、ウェブのデータを扱うhttpは、プロキシというアプリケーションゲートウェイを介した通信をサポートしている。ここでは接続先のIPアドレスやポート番号の情報を記述することができる。しかし、このような方式はすべてのアプリケーションプロトコルが対応しているわけではないの

で、サービスのスケーラビリティが著しく低いといえる。

【0010】これに対して、アプリケーションゲートウェイと連携するクライアントホスト用ソフトウェア（ミドルウェア）を用いる方式がある。これは、すべてのTCPコネクションを一律アプリケーションゲートウェイに向けるもので、一般的なプロキシに比べるとサービスのスケーラビリティが格段に向上している。しかし、以下に示すような、コネクションの確立過程が2段階構成で2回目のコネクションが1回目の逆向きに確立するプロトコルには対応できない。具体的な例として、アプリケーションゲートウェイでデータ転送アプリケーションf t pを使うことを考える。まずクライアントからサーバに対しての「コントロールセッション」を確立することは可能であるが、データ転送を行なう「データセッション」はサーバからクライアントに対して確立しようとして失敗する。これは、クライアントからサーバへの方はストリームが流れるため、最初の「コントロールセッション」は確立する。しかし、コントロールセッションのサーバから見たエンドホストはゲートウェイであるため、サーバは、「データセッション」をゲートウェイに対して確立しようとするが、ゲートウェイはこのセッション用にパッシブオープンしていないために確立失敗となる。

【0011】f t pに限っていえば、クライアントソフトの“passive mode”を利用することによって、この問題を回避できる。しかし、その他にも2段階構成のコネクション確立を行なうプロトコルは存在するし、今後増えていくことを考えると、サービスのスケーラビリティに課題が残るといえる。このように、従来型TCP終端装置を経路上に持つエンドーエンドにおいては、IPパケットもしくはストリームの到達性に問題があり、「エンド-エンドのグローバルコネクティビティの保証」ができない。

【0012】従来型TCP終端装置の問題点の一つである“エンドーエンド情報の非保存”についてまとめると、図2（a）（b）に示すようになる。図2（a）に示すように、通常のIPルータを通過してもIPパケットヘッダ内のIPアドレスおよびポート番号の情報は一切変化しない。しかし、従来型TCP終端装置を通過すると（実際にはIPパケットは一度ストリームに再構成されてから再びパケット化される）、図2（b）に示すように、そのIPアドレスおよびポート番号は新たなエンドホストとしてゲートウェイのIPアドレスおよびポート番号に置き換わる。現在、多くのアプリケーションゲートウェイやプライベートアドレスとグローバルアドレスの変換を行なうNATルータでは、このような問題を回避するために、アプリケーションプロトコル毎に個別の対応をして、サーバからのTCPパケットやUDPパケットを本来のクライアントに送付する仕組みを提供

している。しかし、その他のアプリケーションプロトコルにおいても、最初のセッションから、クライアントのIPアドレスを取得して、その情報を元にサーバからクライアントに対してTCPコネクションを張ることやUDPパケットを送付することは少なくない。さらに、新たなアプリケーションプロトコルの出現に応じて、逐一对応しなければならないという問題があり、サービスに対するスケーラビリティがない方式といえる。

【0013】本発明は、上記事情を考慮してなされたものであって、本発明は、性質の異なるネットワークを跨るTCPコネクションにおいて、効率の良い伝送を行なうために、TCP終端の仕組みを採り入れつつ、インターネットの性質として重要なグローバルコネクティビティを保証したTCPコネクションを終端させる機能を備えたIPルータ装置を実現することである。

【0014】

【課題を解決するための手段】図1は、本発明の概要を説明する図である。同図において、11a、11bはネットワークドライバ、12a、12bはIPスタック、13a、13bはTCPスタック、14はストリーム転送手段、15はTCPコネクション管理データベース、16はIPフォワーディングである。本発明のIPルータ装置の基本部分の構成は従来型のアプリケーションゲートウェイと同様であるが、前記図17に示した従来型のアプリケーションゲートウェイに対して本発明では以下の点が追加・変更されている。

（a）IPスタック12a、12bにおけるTCP/IPパケットヘッダ内情報の変換手段（図1の変換手段1～4）とそれ用のデータベース（図1のTCPコネクション管理データベース15）を設けた点。

（b）ストリーム転送手段14におけるTCP接続先IPアドレスとポート番号の抽出機能を設けた点。

TCP/IPパケットヘッダ内情報変換手段1～4は入力パケットと出力パケットに対する処理としてそれぞれ存在する。これらは、各ネットワークドライバ（ネットワークインターフェース）11a、11bについて個別に動作する。ただし、各変換機能1～4は、共通のTCPコネクション管理データベース15を介して連携し、一つのTCPコネクションに対して一意の変換規則にしたがって処理が実行される。ストリーム転送手段14は、従来型のアプリケーションゲートウェイのそれと比較して、クライアントとのTCPコネクションの情報を元に、TCP接続先アドレスとポート番号の抽出を行なう機能を有する。

【0015】図1に示すように本発明においては、次のようにして前記課題を解決する。

（1）TCPコネクションを終端させる機能を備えた複数の異なるIPネットワークを接続するルータ装置において、該ルータ装置に、中継しようとするTCPコネクションを形成する複数のIPパケットが前記ルータ装置

## 7

を通過しようとする際に、パケット内のIPアドレス情報およびポート番号情報の一部を所定の規則に従い書き換える第1の変換手段(図1の変換機能1、2)と、前記TCPコネクションの本来の接続先を示す情報を抽出し、前記ルータからその接続先へTCPコネクションを生成し、前記2つのTCPコネクションをストリームで連結する手段14と、上記本来の接続先へのTCPコネクションについて、それを形成する複数のIPパケット内のIPアドレス情報およびポート番号情報の一部を所定の規則に従い書き換える第2の変換手段(変換機能

3、4)とを設け、上記第1、第2の変換手段は、上記2つのTCPコネクションを1組として取り扱い、上記1組のTCPコネクションに対して一意の識別番号を付与して管理データベース15に格納し、該データベース15に格納された一意の識別番号により、2つのTCPコネクションを管理する。

(2) 上記(1)において、第1、第2の変換手段は、TCPスタックからのTCPコネクションの終了の通知を受けて、上記データベースに格納している識別番号の情報を作成、消去もしくは、更新する。

(3) 上記(1)または(2)において、第1、第2の変換手段は、TCP以外のIPパケットについては、IPパケット内の情報の書き換えを行わず、所定のネットワークにルーティングする。

(4) 上記(1)(2)(3)において、第1、第2の変換手段は、所定のコネクション数を超えた新たなTCPコネクションのIPパケットについて、IPパケット内の情報の書き換えを行わず、所定のネットワークにルーティングする。

(5) 上記(1)～(4)において、第1、第2の変換手段は、所定の型をパケット内のヘッダ情報として持つICMPパケットについて、そのIPアドレス情報およびポート番号情報、および、パケットデータ内に含まれるTCP/IPのヘッダ情報の一部を書き換える。なお、上記第1、第2の変換手段、ストリームを連結する手段が、TCPコネクションの1組に対してそれぞれ1つの識別番号を付与し、それを複数管理するように構成することもできる。これにより、複数のTCPコネクションのサポートが可能となる。

【0016】図2に本発明と従来型のIPルータ装置、従来型のアプリケーションゲートウェイにおける前後のパケット情報を示す。図2(a)は前記した通常のIPルータを通過した前後のパケットヘッダ内のIPアドレスとポート番号の情報を示す。ここでは、ルータを通過することによって、何らアドレスおよびポート番号情報に変化がないという特徴を持つ。これが、上述の「グローバルコネクティビティの保証」を意味する。一方、図2(b)は前記した通常のアプリケーションゲートウェイを通過した前後のパケットヘッダ内のIPアドレスとポート番号の情報を示す。アプリケーションゲートウ

## 8

イを境にパケットヘッダ内の情報が変化していることがわかる。従って、「グローバルコネクティビティの保証」ができなくなる。本発明では、図2(c)に示すように、TCPコネクションを終端するものの、パケットヘッダ内のIPアドレスとポート番号の情報には変化がない。従って、両エンドホストからは、通常のIPルータと全く同様に、TCPコネクションを実現できる。その結果グローバルコネクティビティについても保証される。

10 【0017】

【発明の実施の形態】図3に本発明を装置として実現した場合の構成例を示す。通常、汎用コンピュータ上の主記憶装置102もしくは補助記憶装置103上に本発明の機能を実現するためのソフトウェアが記憶され、CPU101で本発明の処理が実行される。図3に示す装置には2つ以上のネットワークインターフェース104が搭載され、異なるネットワークを連結したものとなる。なお、ネットワークインターフェースとしては、イーサネットのみならずシリアル回線上のPPPなどを含む。

20 【0018】図4に本発明の実施例のIPルータ装置の構成を示す。本実施例のIPルータ装置は、IPスタック内でTCPコネクションの情報を管理してパケットヘッダの情報を変換する変換手段と、アプリケーション層において、2つのTCPコネクションをストリームレベルで接続する手段に大別される。同図において、2、3はそれぞれホストであり、ここでは同図に示すようにホスト1のIPアドレスはa1、アプリケーションのクライアントポート番号(動的にアサイン)はp1、ホスト2のIPアドレスはa4、アプリケーションサーバのサーバポート番号(アプリケーションサーバ毎に固定値)はp4であるとする。

30 【0019】1は本実施例のIPルータ装置であり、図4に示すようにIPルータのIPアドレスはa2、a3、ストリーム転送ソフトのサーバポート番号(固定値)はp2、ストリーム転送ソフトのクライアントポート番号(動的にアサイン)はp3であるとする。IPルータ装置1はネットワークドライバ11a、11b、IPスタック12a、12b、TCPスタック13a、13b、2つのTCPコネクションをストリームで連結する機能を備えたストリーム転送手段14を備え、ストリーム転送手段14はTCPコネクションの本来の接続先である接続先IPアドレスを抽出する機能を備えている。15はTCPコネクションを管理するデータベースであり、上記IPスタック12a、12bは該データベース15と連携してパケットヘッダの変換を行う。また、IPフォワーディング16は、後述するようにIPスタックにおいて変換対象外となったIPパケットを前記図16に示した従来型ルータ装置と同様にIPルーテ

50 【0020】図5にIPスタック内でパケットヘッダの

変換を行なう機能とその規則、さらにTCPコネクション管理データベースと前記変換機能との連携を示す。同図において、“src a1, p1”の“a1”は送信元のIPアドレス（src-IP），“p1”は送信元のポート番号（src-ポート番号）を示し、“dst a4, p4”の“a4”は接続先のIPアドレス（dst-IP），“p4”は接続先のポート番号（dst-ポート番号）を示し、変換機能1～4により、“src a1, p1”、“dst a4, p4”がそれぞれ「元」→「新」に変換されることを示している。また、

【0021】図5の変換規則はルータを外から見た際には、図3で説明したパケット情報（ヘッダ内のIPアドレスとポート番号）に変化がないことを示し、一方で、ルークのアプリケーション層から見た際には、ストリーム転送機能が2つの終端したTCPコネクションをハンドリングできることを示している。一連の変換機能は以下の機能からなる。

(1) ネットワークドライバ11a, 11bからIPスタック12a, 12bに流入するIPパケットの処理（図6のフロー）。

(2) IPスタック12a, 12bからネットワークドライバ11a, 11bに流出するIPパケットの処理（図7のフロー）。

(3) TCPコネクション管理データベース15（以下管理データベースと略記する）のエントリ削除処理（図8のフロー）。

上記(1)(2)のIPパケットの処理において、IPスタック12a, 12bは図5に示す変換機能1～4により、IPパケットのヘッダ部のIPアドレス、ポート番号を変換する（図9～図12のフロー）。なお、図4においては、便宜上、ルータ内のIPスタックとTCPスタックを2つに分離して記述しているが、実施形態においては、インターフェースが複数あっても、IPスタックとTCPスタックは共通に存在し利用される。従って、上記パケット処理(1)は、変換機能1と変換機能4を含む。同様に、上記パケット処理(2)は、変換機能2と変換機能3を含む。

【0022】従来型IPルータでは、流入するIPパケットの処理は、前記図21に示したように、dst-IPアドレスが自ホスト（ルータ自身）宛かどうかで判断し、自ホスト宛以外のIPパケットをIPフォワーディング経路でネットワークに出力する（IPルーティング）。本実施例では、このIPルーティングの代わりにTCP終端を行なうので、自ホスト宛以外のIPパケットがTCPもしくはICMPならば、図4に示したように変換機能4もしくは変換機能1を通す（後述する図6

のフロー）。それぞれの変換機能では、TCPコネクション管理データベース15に登録されたパケットを対象に、図5に示すようにIPアドレスおよびポート番号の変換を行ない、ルータ内のTCPスタック13aもしくは13bに引き渡す。また、上記管理データベース15に登録されていないパケットは変換対象外とする。なお、変換機能1では、後述するように変換対象外となったIPパケットがTCPのSYNフラグ（通信確立要求フラグ）を持ち、新規にTCPコネクションを確立しようとしている場合は、データベースにエントリを追加する。変換機能1はこの新規エントリ追加機能を有するために、変換機能4よりも後に処理される必要がある。最終的に変換対象外となったIPパケット、つまり、UDPパケットや何らかの制限（後述のコネクション数制限など）に引っかかったTCPパケットは、従来型のIPルータと同様にIPフォワーディング16経由でIPルーティングする。

【0023】また、流出するIPパケットの処理は、従来型IPルータでは前記図22に示したようにdst-IPアドレスに従ったネットワークをルーティングテーブルから検索し、そのネットワークに出力する。本実施例では、IPフォワーディング16経由で送られてきたパケット以外つまり、アプリケーション層からTCP/ICMP層に引き渡されてきたIPパケットを対象に、変換機能2もしくは変換機能3に渡し、図5に示すようにIPアドレスおよびポート番号の変換を行う。変換後は従来型IPルータと同様の処理を行ないネットワークに出力する。

【0024】TCPコネクション管理用データベース15は、TCPコネクションをその生成から終了までの間管理し、コネクション毎に前記で述べた一意の識別番号（図5のpN）を割り振る。上記管理用データベース15で管理するエントリは、図5に示すように項目A～Eの5つからなる。各エントリは、変換機能1における新規TCPコネクションの検出と同時に新規生成され、項目A～Dが割り当てられる。さらに、変換機能3では、該当するTCPコネクションのIPパケットが通過する際に、図5に示すように項目Eが割り当てられる。これによって、変換機能1～4はデータベースのエントリと連携して、パケットヘッダ中のIPアドレスおよびポート番号を書換えることが可能となる。上記管理用データベース15のエントリは、後述する図8のフローに示すように、TCPスタックにおいてTCPコネクションがクローズされるのと同期して削除される。すなわち、TCPコネクションクローズ時、TCPスタックは、TCPコネクションの識別番号をともない上記管理用データベース15内のコネクション終了処理ルーチン呼び出す。上記処理ルーチンでは、識別番号からエントリを検索し、データベースからそれを削除する。

【0025】管理データベース15内でのコネクション



を管理するためのエントリは、上述のとおり、TCPパケットのSYN（通信確立要求）が来たときに新規に作成し、TCPスタックのクローズ処理からの呼び出しを受けて、エントリを削除する。これは、TCPスタックが動作するごとく、コネクションの存在に関わるすべてのTCPシーケンス〔SYN（通信確立要求）、FIN（終了要求）、RST（強制終了）〕を追う必要がないのでコネクションを管理する処理を大幅に簡略化できる。

【0026】ストリーム転送手段14は後述する図13のフローに示すようにストリーム転送を行う。前記図23に示した従来型アプリケーションゲートウェイで 사용되는典型的なストリーム転送機能と、本実施例のストリーム転送機能で異なる点は、接続すべきサーバのIPアドレスとポート番号の取得方法にある。従来型アプリケーションゲートウェイにおいては、クライアントとのストリーム中にその情報を提示する方式が一般的である。例えば、アプリケーションゲートウェイの一つであるhttpプロキシでは、そのストリーム中に以下に示すように“www.nic.ad.jp”というIPアドレスを得るためのホスト情報と“80”というポート番号が記述される。

```
GET/HTTP/1.0（エンドーエンド通信時）
GET http://www.nic.ad.jp:
80/HTTP/1.0（プロキシ使用時）
```

【0027】一方、本実施例では、前記図5に示したように変換機能1の後ろにIPパケット内に本来の接続先の情報（図5における“dst a4”）が残るため、ストリーム転送機能では、ソケットの接続先情報を取得する関数を用いて、接続すべきサーバのIPアドレスを取得できる。なお、ポート番号については、変換機能1において、ストリーム転送機能のサーバソケットのサーバポート（図5の“p2”）に書き換えるために、ストリーム転送機能においては本来の値は取得できない。しかし、図5に示したように変換機能1において、ソケットのsrcポート番号にTCPコネクションの識別番号（PN）が埋め込まれるので、これを、サーバへのTCPコネクションのポート番号として用いる。このポート番号の情報は、TCPパケットが変換機能3を通過する際に、図5に示したように本来の接続先であるサーバのポート番号（図5の“p4”）に書き換えられる。ちなみに、変換機能3では、dstポート番号だけでなく、src-IPアドレス、src-ポート番号も書き換えを行なうので、ストリーム転送機能が動的なポート番号を割り振るクライアントソケットを使用しても、サーバからは常に本来のクライアントの割り振ったIPアドレスとポート番号が見える。

【0028】以下図6～図13のフローチャートにより、上述した処理について説明する。図6は前記（1）のIPスタックにおけるIPパケットの入力処理を示す

図である。IPスタックにネットワークドライバからIPパケットが流入すると、dst-IPアドレスが自ホスト宛であるかを調べ（ステップS1）、自ホスト宛であればIPパケットを自ホストのTCP/IPスタックに引き渡し（ステップS2）処理を終了する。また、dst-IPアドレスが自ホスト宛でない場合には、TCPもしくはICMPパケットであるかを調べ（ステップS3）、TCPもしくはICMPパケットでなければ、IPフォワーディング16経由でIPパケットの出力処理に引き渡す（ステップS4）。

【0029】また、IPパケットがTCPもしくはICMPパケットの場合にはステップ5に行き変換機能4（後述する図12のフロー）によりTCP/IPパケット内のIPアドレス、ポート番号を例えば図5の変換機能4に示すように変換する。そして変換されたか否かを判別し（ステップS6）、変換された場合には自ホストのTCP/IPスタックに引き渡し（ステップS2）処理を終了する。また、変換されなかった場合には、変換機能1（図9のフロー）によりTCP/IPパケット内のIPアドレス、ポート番号を例えば図5の変換機能1に示すように変換する。そして変換されたか否かを判別し（ステップS8）、変換されなかった場合には、IPフォワーディング16経由でIPパケットの出力処理に引き渡す（ステップS4）。変換された場合には、自ホストのTCP/IPスタックに引き渡し（ステップS2）処理を終了する。

【0030】図7は前記（2）のIPスタックにおけるIPパケットの出力処理を示す図である。IPパケットがIPスタックに流入すると、IPパケットがIPフォワーディング16経由で送られてきたものか調べる（ステップS1）。IPパケットがIPフォワーディング16経由で送られてきたものである場合にはステップS6にいき、dst-IPアドレスもしくはそのサブネットワークで、ルーティングテーブルを検索し（ステップS6）、該ルーティングテーブルに従い、ネットワークドライバにIPパケットを引き渡す。IPパケットがIPフォワーディング16経由で送られてきたものでない場合には、TCPもしくはICMPパケットかを調べ（ステップS2）、TCPもしくはICMPパケットでない場合にはステップS6に行き、上記処理を行う。

【0031】IPパケットがTCPもしくはICMPパケットの場合には、変換機能2（図10のフロー）によりTCP/IPパケット内のIPアドレス、ポート番号を例えば図5の変換機能2に示すように変換する（ステップS3）。そして変換されたか否かを判別し（ステップS4）、変換された場合には、ステップS6に行く。またステップ変換機能2により変換されなかった場合には、変換機能3（図11のフロー）によりTCP/IPパケット内のIPアドレス、ポート番号を変換する（ステップS5）。そして、ステップS6に行き、上記のよ

うに `dst-IP` アドレスもしくはそのサブネットワークで、ルーティングテーブルを検索し（ステップ S 6）、該ルーティングテーブルに従い、ネットワークドライバに `IP` パケットを引き渡す。

【0032】図 8 は前記（3）に示した管理データベース 15 のエントリ削除処理を示す図である。TCP スタック内のコネクションが終了すると、TCP スタック内のコネクション終了処理は、管理データベース 15 にコネクションの終了を通知する（ステップ S 1）。これにより管理データベース 15 内のコネクション終了処理が呼び出され、該コネクション終了処理は TCP コネクションの識別番号（前記した“`pN`”）からデータベース内のエントリを検索し（ステップ S 3）、該エントリをデータベースから削除する（ステップ S 4）。一方、TCP スタック内のコネクション終了処理は、TCP コントロールブロックを削除する（ステップ S 2）。

【0033】図 9 は、前記した変換機能 1 における処理を示す図であり、ネットワークドライバから `IP` スタックに流入する TCP/`IP` パケット内の `src-IP` アドレス、`src`-ポート番号、`dst-IP` アドレス、`dst`-ポート番号は変換機能 1 により次のように変換される。まず、ステップ S 1 において、管理データベース 15 内のエントリを検索し、TCP/`IP` パケット内の `src-IP` アドレスが管理データベース 15 内の項目 A（図 5 参照）の `IP` アドレスに一致し、`src`-ポート番号が項目 A のポート番号に一致し、`dst-IP` アドレスが項目 B の `IP` アドレスに一致し、`dst`-ポート番号が項目 B のポート番号に一致するエントリがあるかを調べる。管理データベース 15 内に上記エントリが存在しない場合には、ステップ S 2 からステップ S 3 30 に行き、処理対象となるコネクションがコネクション数の制限内であるかを調べ、制限内でない場合には処理を終了する。この場合には、`IP` フォワーディング 16 経由で `IP` パケットの出力処理に引き渡される。

【0034】コネクション数の制限内である場合には、管理データベース 15 内に新たなエントリを作成し（ステップ S 4）、上記エントリ内の項目 A に `src-IP` アドレスおよび `src`-ポート番号を登録する（ステップ S 5）。また、上記エントリ内の項目 B に `dst-IP` アドレスおよび `dst`-ポート番号を登録する（ステップ S 6）。例えば図 5 においては、項目 A に“`a1, p1`”が、項目 B に“`a4, p4`”が登録される。さらに、上記エントリ内の項目 C、D の `IP` アドレスに項目 A、B の複製を登録する。また、ポート番号に TCP コネクションの識別番号を登録する（ステップ S 7）。例えば図 5 においては、項目 C、D のポート番号に“`pN`”が登録される。

【0035】ついで、TCP/`IP` パケット内の `src`-ポート番号に項目 C のポート番号を設定し（ステップ S 8）、TCP/`IP` パケット内の `dst`-ポート番号

にストリーム転送ソフトのサーバポート番号を設定する（ステップ S 9）。そして、`IP` パケットのチェックサムを計算しなおして（ステップ S 10）、処理を終了する。また、ステップ S 2 において、ステップ S 1 の条件を満たすエントリが存在すると判別された場合には、ステップ S 2 からステップ 8 に行き、上記ステップ S 8 ～ステップ 10 の処理を行う。上記処理により、例えば図 5 においては、同図の変換機能 1 の「新」に示すように、`src-IP` アドレス、`src`-ポート番号にそれぞれ `a1, pN` が設定され、`dst-IP` アドレス、`dst`-ポート番号にそれぞれ `a4, p2` が設定される。

【0036】図 10 は、前記した変換機能 2 における処理を示す図であり、`IP` スタックからネットワークドライバに流出する TCP/`IP` パケット内の `src-IP` アドレス、`src`-ポート番号、`dst-IP` アドレス、`dst`-ポート番号は変換機能 2 により次のように変換される。まず、ステップ S 1 において、管理データベース 15 内のエントリを検索し、TCP/`IP` パケット内の `dst-IP` アドレスが管理データベース 15 内の項目 C の `IP` アドレスに一致し、`dst`-ポート番号が項目 C のポート番号に一致するエントリが存在するかを調べる。

【0037】管理データベース 15 内に上記エントリが存在しない場合には、処理を終了する。また、上記エントリが存在する場合には、ステップ S 2 からステップ S 3 30 に行き、TCP/`IP` パケット内の `src`-ポート番号に項目 B のポート番号を設定し（ステップ S 3）、TCP/`IP` パケット内の `dst`-ポート番号に項目 A のポート番号を設定する（ステップ S 4）。そして、`IP` パケットのチェックサムを計算しなおして（ステップ S 5）、処理を終了する。上記処理により、例えば図 5 においては、同図の変換機能 2 の「新」に示すように、`src-IP` アドレス、`src`-ポート番号にそれぞれ `a4, p4` が設定され、`dst-IP` アドレス、`dst`-ポート番号にそれぞれ `a1, p1` が設定される。

【0038】図 11 は、前記した変換機能 3 における処理を示す図であり、`IP` スタックからネットワークドライバに流出する TCP/`IP` パケット内の `src-IP` アドレス、`src`-ポート番号、`dst-IP` アドレス、`dst`-ポート番号は変換機能 3 により次のように変換される。まず、ステップ S 1 において、管理データベース 15 内のエントリを検索し、TCP/`IP` パケット内の `dst-IP` アドレスが管理データベース 15 内の項目 D の `IP` アドレスに一致し、`dst`-ポート番号が項目 D のポート番号に一致するエントリが存在するかを調べる。

【0039】管理データベース 15 内に上記エントリが存在しない場合には、処理を終了する。また、上記エントリが存在する場合には、ステップ S 2 からステップ S 3 30 に行き、TCP/`IP` パケット内の `src-IP` アド

レスに項目AのIPアドレスを設定し(ステップS3)、TCP/IPパケット内のsrcポート番号に項目Aのポート番号を設定し(ステップS4)、TCP/IPパケット内のdstポート番号に項目Bのポート番号を設定する(ステップS5)。そして、IPパケットのチェックサムを計算しなおす(ステップS6)。ついで、上記エントリが項目Eに登録済であるかを調べ(ステップS7)、登録されていない場合は、上記エントリの項目Eにsrc-IPアドレス、srcポート番号を登録して(ステップS8)処理を終了する。上記処理により、例えば図5においては、同図の変換機能3の「新」に示すように、src-IPアドレス、srcポート番号にそれぞれa1, p1が設定され、dst-IPアドレス、dstポート番号にそれぞれa4, p4が設定される。また、管理データベース15の項目Eに「a3, p3」が登録される。

【0040】図12は、前記で述べた変換機能4における処理を示す図であり、ネットワークドライバからIPスタックに流入するTCP/IPパケット内のヘッダ部のsrc-IPアドレス、srcポート番号、dst-IPアドレス、dstポート番号は変換機能4により次のように変換される。まず、ステップS1において、管理データベース15内のエントリを検索し、TCP/IPパケット内のsrc-IPアドレスが管理データベース15内の項目BのIPアドレスに一致し、srcポート番号が項目Bのポート番号に一致し、dst-IPアドレスが項目AのIPアドレスに一致し、dstポート番号が項目Aのポート番号に一致するエントリがあるかを調べる。管理データベース15内に上記エントリが存在しない場合には、処理を終了する。また、上記エントリが存在する場合には、ステップS2からステップS3に行き、TCP/IPパケット内のsrcポート番号に項目Dのポート番号を設定し(ステップS3)、TCP/IPパケット内のdst-IPアドレスに項目EのIPアドレスを設定し(ステップS4)、TCP/IPパケット内のdstポート番号に項目Eのポート番号を設定する(ステップS5)。そして、IPパケットのチェックサムを計算しなおし(ステップS6)、処理を終了する。上記処理により、例えば図5においては、同図の変換機能4の「新」に示すように、src-IPアドレス、srcポート番号にそれぞれa4, pNが設定され、dst-IPアドレス、dstポート番号にそれぞれa3, p3が設定される。

【0041】図13は前記で述べたストリーム転送機能の処理フローを示す図である。本実施例におけるストリーム転送機能は、前記図23に示した従来型ストリーム転送機能と同様であるが、前述したようにソケットの接続先情報を取得する関数を用いて接続すべきサーバのIPアドレスを取得する。すなわち、図13のフローに示すように、サーバソケットとしてTCPコネクションの

接続を待ち(ステップS1)、クライアントとTCPコネクションを確立する(ステップS2)。ついで、クライアントルータ間のTCPコネクション情報からdst-IPアドレスおよびsrcポート情報を取得し(ステップS3)、クライアントソケットとして、上記接続先にTCPコネクションを確立する(ステップS4)。そして、ストリームが継続しているかを判断し(ステップS5)、継続していない場合には処理を終了する。また、継続している場合には、クライアントとのストリームから一定量以下のデータを読み出し(ステップS6)、サーバとのストリームへ前記データを書き出す(ステップS7)。

【0042】上述の説明は、基本的にはひとつのエンドーエンドのTCPコネクションを実現するための実施例についてであるが、データベースのエントリを複数管理し、TCPコネクションに割り当てる識別番号を管理TCPコネクション線数内で一意に割り振ることによって、複数のTCPコネクションのサポートが実現可能である。TCPコネクション識別番号は、初期値(1024など)から1ずつ増加させ、管理TCPコネクション数の範囲を超えた場合、所定の値で初期値に戻すことによって実現できる。この際に、使用しようとする識別番号が使用中ならばさらに1増加させる。

【0043】従来型のアプリケーションゲートウェイでは、IP層でのパケットルーティングを行なわない。そのために、UDPやICMPなどTCP以外のIPパケットについては、専用の処理を必要としていた。本実施例のルータ装置においては、前述した図6、図7に示したようにTCPパケットだけ(ICMPパケットについては後述する)をルータ内のTCP層に引き渡す仕組みを備えるので、それ以外のIPパケットは従来型IPルータと同様にIPルーティングされる。従って、TCPコネクション以外についてもグローバルコネクティビティが保証される。

【0044】また、従来型のアプリケーションゲートウェイでは、TCPの制御用にバッファを必要とするので、コネクション数が増大するに比例して、メモリ使用量も増大する。これは、通常のIPルーティングに比べると大きなものなので、無制限にTCP終端することはシステムのメモリを多量に消費することにつながり、装置の価格を上昇させるデメリットを生じさせる。本実施例のルータ装置においては、前記図9で説明した変換機能1に示すように、データベースにおいてsrc-IPアドレス毎にコネクション数の管理をしている。そして、新規コネクションに対するエントリを作ろうとするとき、あらかじめ決めてある制限数と現在管理しているコネクション数を比較し、制限数を超えた場合は新たなエントリを作成せず、その後のIPパケットはすべて、IP層でのIPフォワーディング対象となる。これは、コネクション数の一時的あるいは定常的な増加に伴うシス

テムの性能低下や停止を防ぐ効果をもたらす。さらに、装置を設計する上で必要なメモリ量やCPU能力を規定することも可能となる。

【0045】ところで、本実施例においては、所定のICMPパケットについても本実施例のルータ装置内で終端させるようにした。ここで、所定のICMPパケットとは、そのデータ部にTCPヘッダを含むもので、具体的には、前記図20に示したようにICMPヘッダタイプが、3～5、11、12が対象となる。すなわち、ICMPパケットは前記図20に示したようなパケットヘッダを持ち、そのデータ部にIPヘッダとその上位層

(TCPなど)ヘッダを格納して送る。ここで、図14に示すように、ホストAから本実施例のルータ装置を通過し、別のルータもしくは終点のホストの到達した際に何らかのエラーが生じた場合、ICMPパケットによってそのエラーをパケット送信ホストに通知することがある。

【0046】その際は、ICMPパケットのデータ部にエラーの原因となったIPパケットの一部を直接挿入する。TCPコネクションを構成するIPパケットに生じたエラーについては、実際のTCPコネクションの終端ホスト(ここでは、本実施例のルータ装置)に対して、通知されるべきである。なぜならば、ICMPデータ部に挿入されたTCPヘッダに含まれるシーケンス番号の情報が本実施例のルータ装置で区切られた2つのTCPコネクションで異なり、ICMPでの通知によって情報の不整合が生じてしまうからである。しかし、本実施例のルータ装置では両エンドのホストに対して、グローバルコネクティビティを保証する動作をしているため、ICMPパケットを生成するホスト(もしくはルータ装置)は、図14のホストAに送付しようとしてしまう。

【0047】そこで、本発明では、TCPコネクションだけではなくて、所定のICMPパケットについても、本発明ルータ内で終端するようにした。ICMPパケットを終端する(すなわち、ヘッダ変換する)ためには通常のTCPパケットと同様にコネクションによるエントリ検索が必要である。ただし、図14に示すようにICMPパケットの進行方向とそのデータ部に含むTCPパケットの進行方向は逆であるので、データ部のIPおよびTCPヘッダ内のsrc情報とdst情報を反転した上で、検索および変換を行なう必要がある。更に、変換後はそれらを元に戻す。また、この変換にともない、ICMPパケット包含するIPパケットのヘッダにおけるdst-IPアドレスも、前記ICMPデータ部に含むIPヘッダの反転後のdst-IPアドレスと同様に変換を行なう。

【0048】ICMPパケットに対する処理を図15に示す。図15において、まず、ICMPパケット内ペイロード(データ部)に格納されているIPパケットおよびTCPパケットのsrcとdstの内容を反転させる

(ステップS1)。ついで、上記ICMP内のペイロード(データ部)内のIPパケットおよびTCPパケットのIPアドレスおよびポート番号の情報を元に管理データベース15のエントリを検索し、さらに、これらの情報におけるICMPを包含するIPパケットのdst-IPアドレスの情報について、変換機能4による変換を行う(ステップS2)。そして、ICMPパケット内のペイロード(データ部)に格納されているIPパケットおよびTCPパケットのsrcとdstの内容を反転させる。

【0049】

【発明の効果】以上説明したように、本発明によれば、性質の異なるネットワークを跨るTCPコネクションにおいて、効率の良い伝送を行なうために、TCP終端の仕組みを採り入れつつ、インターネットの性質として重要なグローバルコネクティビティを保証したIPルータ装置(アプリケーションゲートウェイ)を実現することが可能となる。

【図面の簡単な説明】

【図1】本発明の概要を説明する図である。

【図2】ルータ通過前後のIPパケットの情報変化とグローバルコネクティビティを説明する図である。

【図3】本発明が適用される装置の構成例を示す図である。

【図4】本発明の実施例のルータ装置の構成を示す図である。

【図5】本発明の実施例のルータ装置におけるIPパケット内情報の変換規則を説明する図である

【図6】本発明の実施例のIPスタックにおける入力処理フローを示す図である。

【図7】本発明の実施例のIPスタックにおける出力処理フローを示す図である。

【図8】本発明の実施例の管理データベースのエントリ一削除処理フローを示す図である。

【図9】本発明の実施例の変換機能1の処理フローを示す図である。

【図10】本発明の実施例の変換機能2の処理フローを示す図である。

【図11】本発明の実施例の変換機能3の処理フローを示す図である。

【図12】本発明の実施例の変換機能4の処理フローを示す図である。

【図13】本発明の実施例のストリーム転送機能の処理フローを示す図である。

【図14】IPパケットの生成とペイロード情報の逆転を説明する図である。

【図15】本発明の実施例のICMPパケットの変換処理フローを示す図である。

【図16】従来型IPルータ装置の構成を示す図である。

【図 17】従来型 TCP 終端装置の構成を示す図である。

【図 18】IP パケットのヘッダ構成を示す図である。

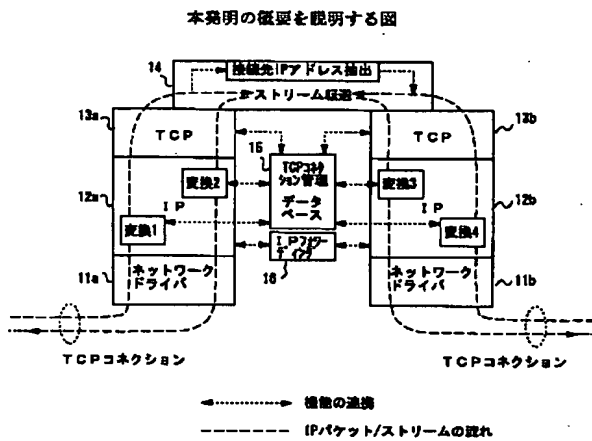
【図 19】TCP パケットのヘッダ構成を示す図である。

【図 20】ICMP パケットのヘッダ構成を示す図である。

【図 21】従来型の IP ルータ装置における IP パケットの入力処理フローを示す図である。

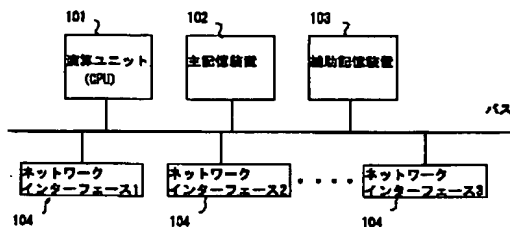
【図 22】従来型の IP ルータ装置における IP パケットの入力処理フローを示す図である。

【図 1】



【図 3】

本発明が適用される装置の構成例を示す図

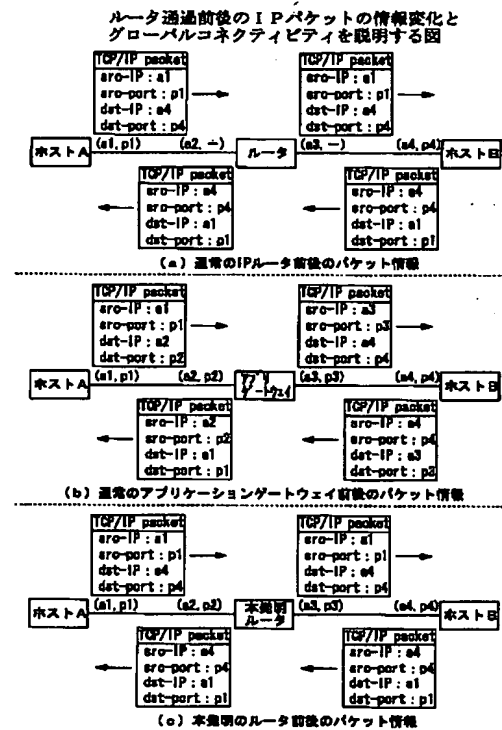


【図 23】従来型の TCP 終端装置におけるストリーム転送機能のフローチャートを示す図である。

【符号の説明】

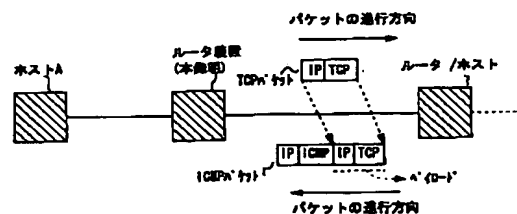
- 1 IP ルータ装置
- 11 a, 11 b ネットワークドライバ
- 12 a, 12 b IP スタック
- 13 a, 13 b TCP スタック
- 14 ストリーム転送手段
- 15 TCP コネクション管理データベース
- 16 IP フォワーディング

【図 2】



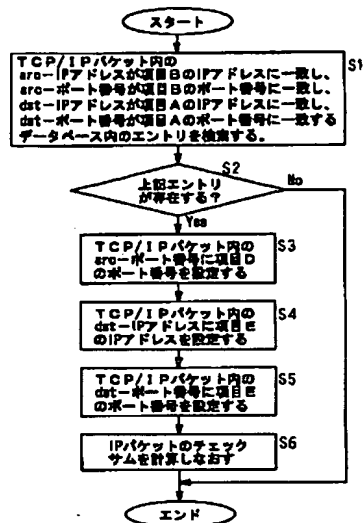
【図 14】

IP パケットの生成とペイロード情報の逆転を説明する図



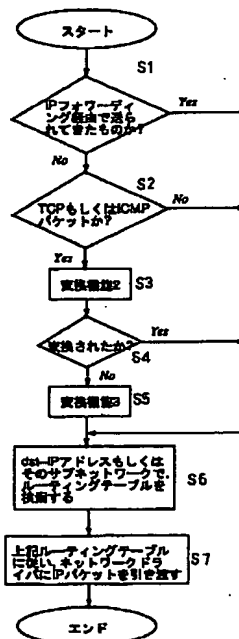
【图 12】

本発明の実施例の変換機能４の処理フローを示す図



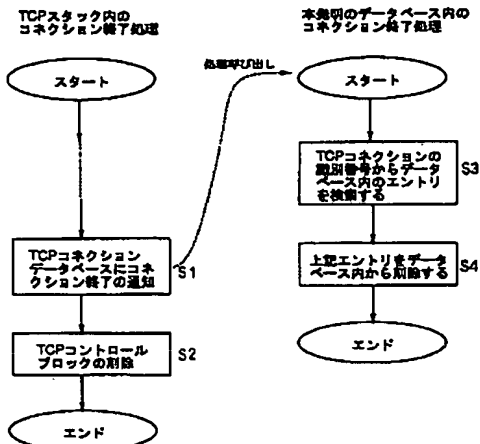
【図 6】

本発明の実施例のＩＰスタックにおける入力処理フローを示す図



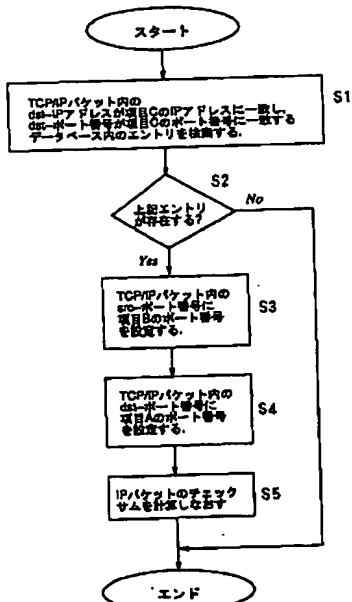
【図 8】

本発明の実施例の管理データベースのエントリ削除処理フローを示す図



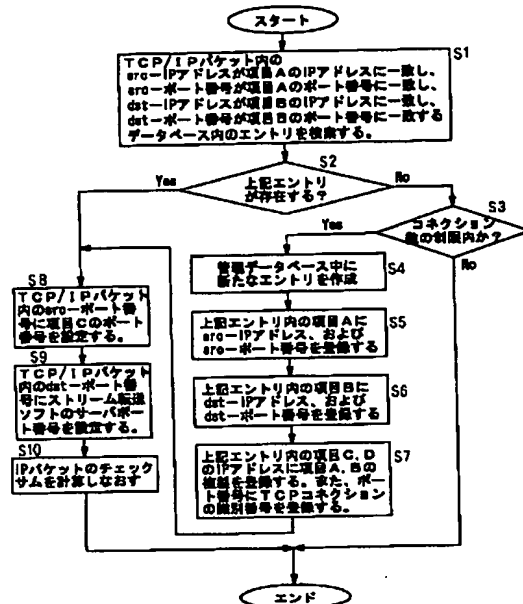
【図 10】

本発明の実施例の変換機能 2 の処理フローを示す図



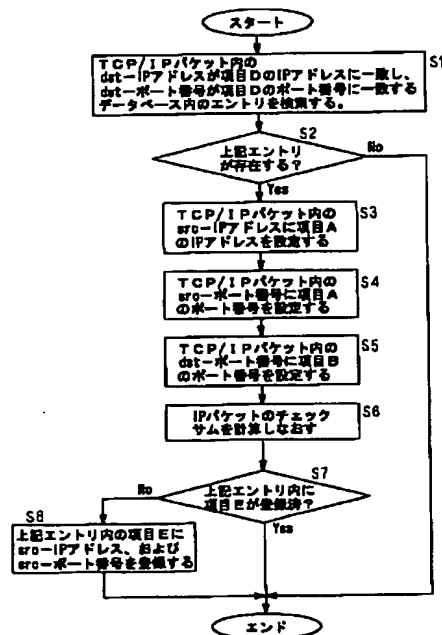
【図 9】

本発明の実施例の変換機能 1 の処理フローを示す図



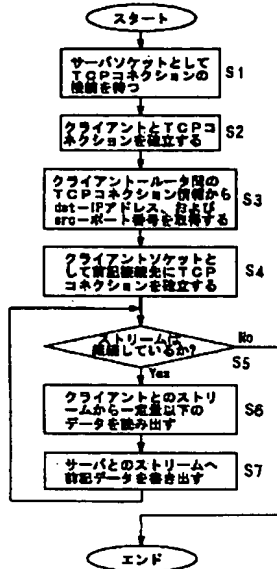
【図 11】

本発明の実施例の変換機能 3 の処理フローを示す図



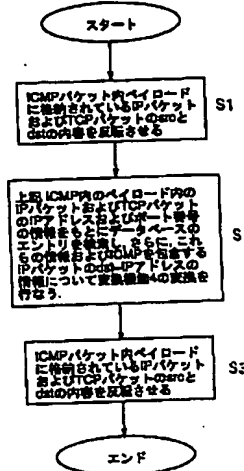
【図13】

本発明の実施例のストリーム転送機能の処理フローを示す図



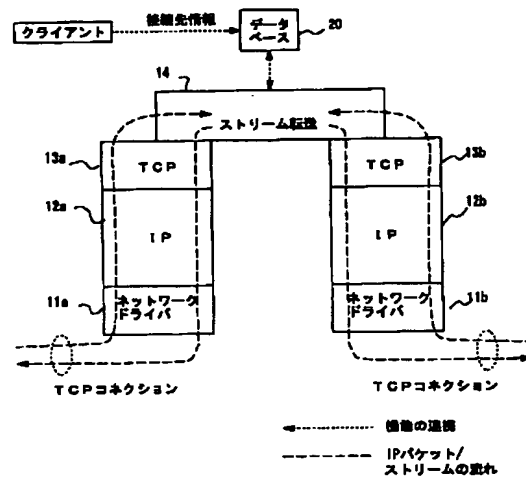
【図15】

本発明の実施例のICMPパケットの変換処理フローを示す図



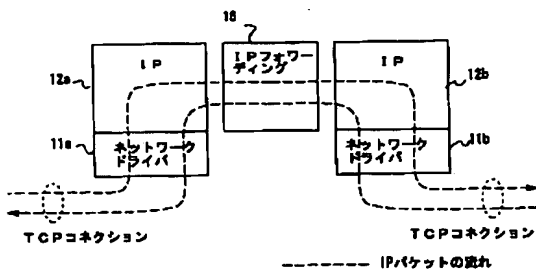
【図17】

従来型TCP終端装置の構成を示す図



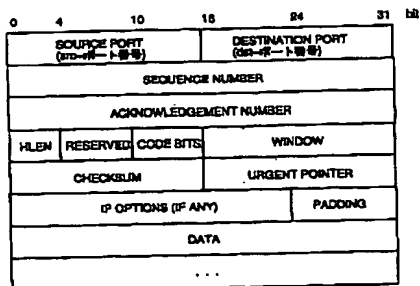
【図16】

従来型IPルータ装置の構成を示す図



【図19】

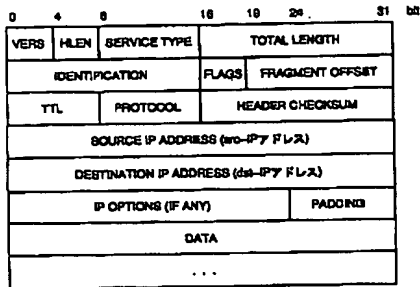
TCPパケットのヘッダ構成を示す図





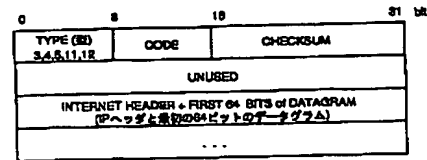
【図18】

IPパケットのヘッダ構成を示す図



【図20】

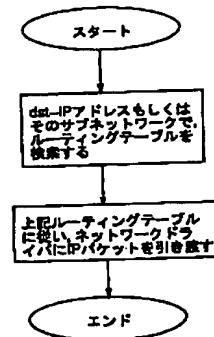
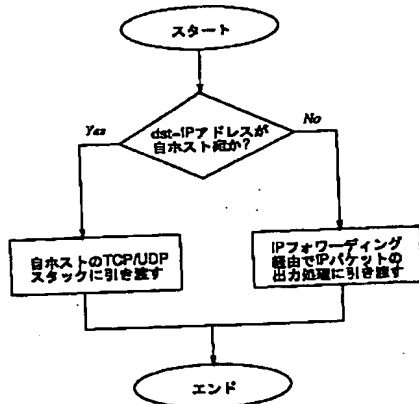
ICMPパケットのヘッダ構成を示す図



【図22】

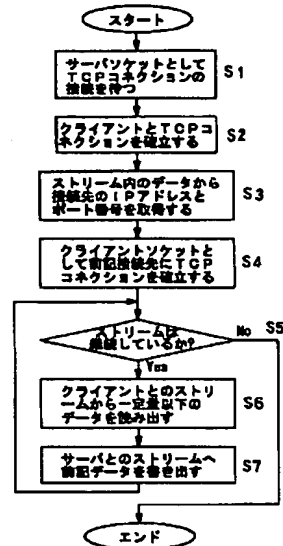
従来型のIPルーティング装置におけるIPパケットの出力処理フローを示す図

従来型のIPルーティング装置におけるIPパケットの入力処理フローを示す図



【図23】

従来型のTCP終端装置におけるストリーム転送機能の  
フローチャートを示す図



フロントページの続き

(51) Int. Cl. 7

識別記号

F I

テーマコード\* (参考)

H O 4 M 3/00

(72) 発明者 中川 格  
神奈川県川崎市中原区上小田中4丁目1番  
1号 富士通株式会社内

(72) 発明者 竹間 智  
神奈川県川崎市中原区上小田中4丁目1番  
1号 富士通株式会社内

(72) 発明者 藤野 信次  
神奈川県川崎市中原区上小田中4丁目1番  
1号 富士通株式会社内

(72) 発明者 谷口 徹哉  
東京都港区虎ノ門二丁目10番1号 エヌ・  
ティ・ティ移動通信網株式会社内

(72) 発明者 久永 隆則  
東京都港区虎ノ門二丁目10番1号 エヌ・  
ティ・ティ移動通信網株式会社内

(72) 発明者 近田 倫康  
東京都港区虎ノ門二丁目10番1号 エヌ・  
ティ・ティ移動通信網株式会社内

(72) 発明者 桑田 大介  
東京都港区虎ノ門二丁目10番1号 エヌ・  
ティ・ティ移動通信網株式会社内

F ターム (参考) 5B089 GA31 GB01 HB02 KA05 KA07  
KB06 KG04 KG08 KH03  
5K030 HA08 HB19 HD03 HD09 KA07  
5K033 CB08 CC01 DB16 DB18 EC03  
5K051 AA04 BB02 CC02 DD01 EE01  
EE02 FF16 GG01 HH17 HH27  
JJ04  
9A001 BB06 CC06 HH34 JJ25 KK56